IN THE CLAIMS:

The text of all pending claims, (including withdrawn claims) is set forth below. Cancelled and not entered claims are indicated with claim number and status only. The claims as listed below show added text with underlining and deleted text with ~~strikethrough~~. The status of each claim is indicated with one of (original), (currently amended), (cancelled), (withdrawn), (new), (previously presented), or (not entered).

1. (canceled)

2. (currently amended) The ~~program~~ method according to claim ~~1~~ 6, wherein the LU decomposition is executed in parallel by each processor of each node in a recursive procedure.

3. (currently amended) The ~~program~~ method according to claim ~~1~~ 6, wherein in said update step, while computing a row block, each node transfers data that belongs to a computed block and is needed to update other blocks, to other nodes in parallel to the computation.

4. (currently amended) The ~~program~~ method according to claim ~~1~~ 6, wherein said parallel computer is a SMP node distributed-memory type parallel computer in which each node is a SMP (symmetric multi-processor).

5. (currently amended) A matrix processing device of a parallel computer in which a plurality of processors and a plurality of nodes including memory are connected through a network, comprising:

a first allocation unit ~~distributing and allocating one combination of bundles of row blocks of a matrix, cyclically allocated, to each node in order to process the combination of the bundles~~ dividing an array A(1:k,1:k) of a matrix to be processed by the number n of nodes and assigning divided matrices subarrays A(1:k/n,1:k), ... , A(k(n-1)/n:k, 1:k), dividing one of the subarrays into narrow block by integer m, and each node's reading data from memory so that a first narrow block is placed in a first node, a second narrow block is placed in a second node, ... , a m-th narrow block is placed in a (mod(m-1,n)+1)-th node;

a separation unit ~~separating a combination of bundles of blocks into a diagonal block, a~~

~~column block under the diagonal block and other blocks~~ eliminating data corresponding to diagonal blocks A(nbase:nbase+m, nbase:nbase+m), where nbase and m are intergers, from data of the narrow blocks placed in each node, at each node;

a second allocation unit redundantly allocating same data as the diagonal block which is eliminated at each node to each node commonly ~~and also allocating one of blocks obtained by one-dimensionally dividing the column block, to each of the plurality of nodes while communicating in parallel~~;

an LU decomposition unit applying LU decomposition to both the diagonal block and the allocated block in parallel in each node ~~while communicating among nodes~~; and

an update unit updating the ~~other~~ blocks of the matrix which is not yet LU-decomposed, using an LU-decomposed block, at each node.

6. (currently amended) A matrix processing method of a parallel computer in which a plurality of processors and a plurality of nodes including memory are connected through a network, comprising:

~~distributing and allocating one combination of bundles of row blocks of a matrix, cyclically allocated, to each node in order to process the combination of the bundles~~ dividing an array A(1:k,1:k) of a matrix to be processed by the number n of nodes and assigning divided matrices subarrays A(1:k/n,1:k), … , A(k(n-1)/n:k, 1:k), dividing one of the subarrays into narrow block by integer m, and each node's reading data from memory so that a first narrow block is placed in a first node, a second narrow block is placed in a second node, … , a m-th narrow block is placed in a (mod(m-1,n)+1)-th node;

~~separating a combination of bundles of blocks into a diagonal block, a column block under the diagonal block and other blocks~~ eliminating data corresponding to diagonal blocks A(nbase:nbase+m, nbase:nbase+m), where nbase and m are intergers, from data of the narrow blocks placed in each node, at each node;

redundantly allocating same data as the diagonal block which is eliminated at each node to each node commonly ~~and also allocating one of blocks obtained by one-dimensionally dividing the column block, to each of the plurality of nodes while communicating in parallel~~;

applying LU decomposition to both the diagonal block and the allocated block in parallel in each node ~~while communicating among nodes~~; and

updating the ~~other~~ blocks of the matrix which is not yet LU-decomposed, using an LU-decomposed block, at each node.

7. (currently amended) A computer-readable storage medium on which is recorded a program for enabling a computer to realize a matrix processing method ~~of~~ as a parallel computer in which a plurality of processors and a plurality of nodes including memory are connected through a network, the method comprising:

~~distributing and allocating one combination of bundles of row blocks of a matrix, cyclically allocated, to each node in order to process the combination of the bundles~~ dividing an array A(1:k,1:k) of a matrix to be processed by the number n of nodes and assigning divided matrices subarrays A(1:k/n,1:k), … , A(k(n-1)/n:k, 1:k), dividing one of the subarrays into narrow block by integer m, and each node's reading data from memory so that a first narrow block is placed in a first node, a second narrow block is placed in a second node, … , a m-th narrow block is placed in a (mod(m-1,n)+1)-th node;

~~separating a combination of bundles of blocks into a diagonal block, a column block under the diagonal block and other blocks~~ eliminating data corresponding to diagonal blocks A(nbase:nbase+m, nbase:nbase+m), where nbase and m are intergers, from data of the narrow blocks placed in each node, at each node;

redundantly allocating same data as the diagonal block which is eliminated at each node to each node commonly ~~and also allocating one of blocks obtained by one-dimensionally dividing the column block, to each of the plurality of nodes while communicating in parallel~~;

applying LU decomposition to both the diagonal block and the allocated block in parallel in each node ~~while communicating among nodes~~; and

updating the ~~other~~ blocks of the matrix which is not yet LU-decomposed, using the LU-decomposed block, at each node.